

Verifica di TPSIT, classe 5^A ROB.

Nota: le domande sono tutte a risposta aperta, per cui ogni risposta deve essere motivata con COMPLETEZZA. Disegna schemi o diagrammi quando necessari.

1. Disegnare la pila TCP/IP indicando il nome di ciascun livello e il nome dei messaggi per i protocolli di ciascun livello. Descrivere brevemente le funzionalità di ciascuno dei livelli TCP/IP.

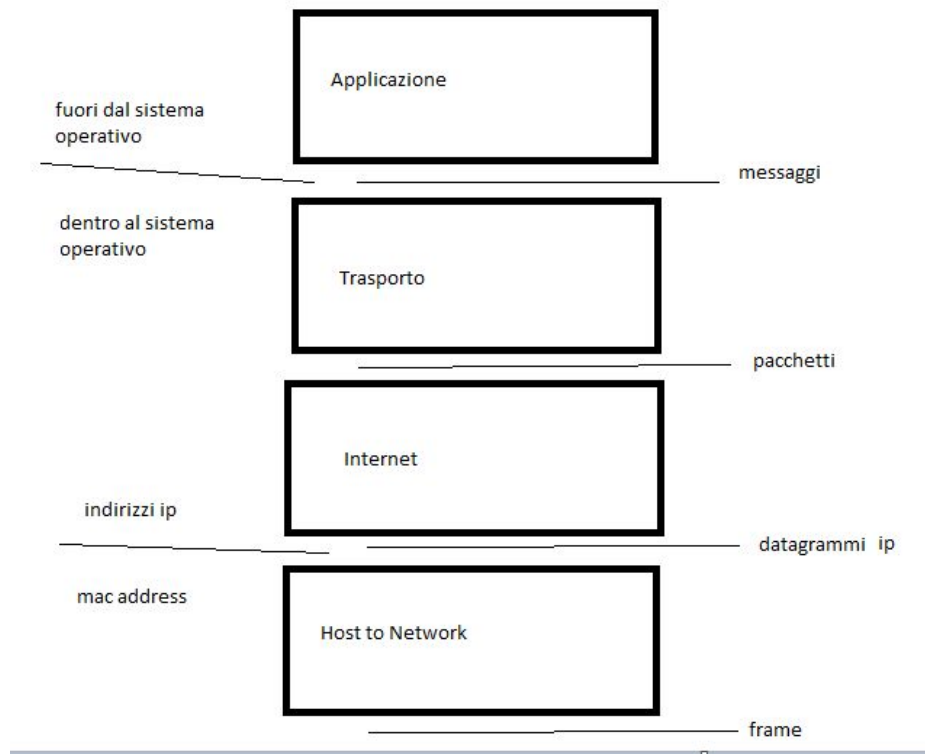
Tracciare e spiegare le due interfacce di separazione presenti nella pila TCP/IP.

2. TCP Port Scanner: uno scanner di porte TCP è un programma che prende di mira un host provando ad effettuare connessioni TCP su tutte le porte possibili per individuare quelle aperte. Una porta è considerata aperta qualora la connessione vada a buon fine.

Scrivere uno scanner di porte TCP in Python 3. Utilizzare il metodo `connect_ex` della classe `socket`: questo metodo è identico al metodo `connect` con la differenza che ritorna 0 in caso di connessione andata a buon fine, mentre se la connessione fallisce ritorna un numero diverso da 0 e NON genera alcun errore.

3. Spiegare il concetto di moltiplicazione/demoltiplicazione.

1.



Il livello applicazione si trova fuori dal sistema operativo ed è per esempio il nostro programma python. Ogni programma utilizza il metodo di trasporto che vuole, pacchetti o flussi di dati. Usa le primitive berkley del livello trasporto.

Il livello trasporto si occupa di mettere in comunicazione due processi. Regola i flussi di dati. Due esempi di protocolli di questo livello sono TCP e UDP. Trasporta i dati senza errori, con la giusta sequenza, in modo affidabile.

Il livello internet si occupa della connessione tra due macchine (M2M). E' suo il compito di decidere se il datagramma che arriva è per lui e per un altro pc. Si salva le tabelle di routing. Fino a questo livello si usano gli indirizzi ip. Controlla anche la validità dei datagrammi.

Il livello host to network è il più basso e si interfaccia con il livello fisico. In questo livello si usano i mac address. Da questo livello escono frame ethernet o wifi, dipende dal tipo di interfaccia che si ha. Si occupa della trasmissione dei datagrammi.

Data: 22 Ottobre 2020



Le due interfacce di separazione sono una a Livello Applicazione che si interfaccia con l'utente e il livello Host to Network che si interfaccia con il mezzo fisico di trasmissione.

2. ...

```
import socket as sock

c = sock.socket(sock.AF_INET,sock.SOCK_STREAM)

# indirizzo ip casuale

ip = "192.168.77.8"

# uso 4000 e 5000 come numeri delle porte solamente come esempio

# porta da cui voglio iniziare a scansionare

portaScan = 4000

# porta fino a cui voglio arrivare a scansionare

portaMax = 5000

# salvo tutte le porte aperte in modo da avere un risultato alla fine

porteAperte = []

while portaScan<portaMax

    if(c.connect_ex((ip,porta)) == 0):

        print(f"la porta {porta} e' aperta")

        porteAperte.append(porta)

        portaScan+=1

c.close()
```

```
print(f"queste sono le porte aperte {porteAperte}")
```

3. Il concetto di demultiplazione e multiplazione consiste nella capacità del livello internet di manipolare qualunque tipo di datagramma contenente un pacchetto TCP o UDP. Cioè ha un procedura specifica per manipolare ciascun tipo di protocollo del livello trasporto. Sia nella demultiplazione in cui vengono inseriti in un datagramma ip o multiplazione quando vengono estratti dal datagramma ip. La demultiplazione avviene da parte di chi invia e la multiplazione da parte di chi riceve.